



UNIVERSITATEA DIN  
BUCUREŞTI



FACULTATEA DE  
MATEMATICA ŞI  
INFORMATICA

SPECIALIZAREA INFORMATICĂ

Lucrare de licență

# Aplicație web pentru personalizarea rețetelor alimentare

Absolvent  
Duca Cosmina-Elena

Coordonator științific  
Conf. Univ. Dr. Boriga Eugen Radu

Bucureşti, iunie 2024

## **Rezumat**

Această lucrare are ca scop prezentarea procesului de dezvoltare al aplicației web "CookIt". Aplicația a fost realizată pentru a facilita gestionarea alimentelor și educarea consumatorilor în vederea reciclării. "CookIt" se adresează utilizatorilor care doresc să economisească bani și resurse, reducând risipa alimentară prin utilizarea strictă a ingredientelor pe care le au la dispoziție.

Aplicația utilizează tehnologii avansate, cum ar fi ChatGPT pentru generarea automată a rețetelor și DALL-E 3 pentru generarea imaginilor cu preparate, oferind astfel o interacțiune plăcută și personalizată. Prin integrarea API-ului YouTube, aplicația oferă utilizatorilor acces la cele mai recente videoclipuri despre prevenirea aruncării alimentelor, contribuind la educarea acestora și la procesul de conștientizare.

Funcționalitățile suplimentare includ gestionarea listelor de cumpărături, ceea ce poate face utilizatorii să fie mai organizați atunci când își achiziționează alimentele. În plus, aplicația permite utilizatorilor să localizeze centre de reciclare pentru materiale specifice, folosind API-ul Google Maps.

"CookIt" oferă utilizatorilor o experiență intuitivă și plăcută, integrând funcționalități variate care răspund atât nevoilor practice, cât și celor educaționale. În ceea ce privește dezvoltarea pe viitor a aplicației, se are în vedere, printre altele, îmbunătățirea algoritmilor de generare a rețetelor pentru a oferi recomandări și mai precise și personalizate. O altă funcționalitate de dezvoltat ar fi adăugarea unei părți de socializare în aplicație, unde utilizatorii pot face schimb unii cu ceilalți de rețete culinare și de idei.

## Abstract

The purpose of this paper is to present the development process of the web application "CookIt". The application was designed to facilitate food management and educate consumers about recycling. "CookIt" targets users who wish to save money and resources by reducing food waste through the use of available ingredients.

The application employs advanced technologies, such as ChatGPT for automatic recipe generation and DALL-E 3 for creating images of dishes, thus offering a pleasant and personalized interaction. Through the integration of the YouTube API, the application provides users with access to the latest videos on preventing food waste, contributing to their education and awareness process.

Additional functionalities include managing shopping lists, which can help users be more organized when purchasing their groceries. Moreover, the application allows users to locate recycling centers for specific materials using the Google Maps API.

"CookIt" offers users an intuitive and enjoyable experience, incorporating a variety of features that meet both practical and educational needs. Future development plans for the application include, among other things, improving recipe generation algorithms to provide even more accurate and personalized recommendations. Another functionality to be developed is the addition of a social component in the app, where users can exchange recipes and ideas with each other.

# Cuprins

<b>1 Introducere</b>	<b>6</b>
1.1 Context . . . . .	6
1.2 Motivație . . . . .	7
1.3 Scopul aplicației . . . . .	7
1.4 Aplicații similare . . . . .	7
<b>2 Tehnologii utilizate</b>	<b>9</b>
2.1 ASP.NET Core . . . . .	9
2.2 Bootstrap . . . . .	9
2.3 ASP.NET Identity . . . . .	10
2.4 Entity Framework Core . . . . .	11
2.5 SQL Server . . . . .	12
2.6 ChatGPT API . . . . .	12
2.7 YouTube Data API . . . . .	15
2.8 Google Maps API . . . . .	17
<b>3 Implementarea aplicației</b>	<b>19</b>
3.1 Baza de date . . . . .	19
3.2 Model-View-Controller . . . . .	20
3.2.1 Models . . . . .	21
3.2.2 Views . . . . .	22
3.2.3 Controllers . . . . .	23
3.3 Funcționalitățile aplicației . . . . .	26
3.3.1 Autentificarea . . . . .	26
3.3.2 Generarea Rețetelor . . . . .	26
3.3.3 Salvarea Rețetelor . . . . .	27
3.3.4 Gestionarea Listei de Cumpărături . . . . .	27
3.3.5 Accesare Centrelor de Reciclare . . . . .	27
3.3.6 Vizualizarea Știrilor . . . . .	28

<b>4 Prezentarea aplicației</b>	<b>29</b>
4.1 UI/UX . . . . .	29
4.2 Interfața . . . . .	30
4.2.1 Pagina de Autentificare . . . . .	30
4.2.2 Pagina Principală . . . . .	31
4.2.3 Pagina pentru Generare . . . . .	31
4.2.4 Pagina Rețetei . . . . .	32
4.2.5 Pagina de Știri . . . . .	32
4.2.6 Pagina Listelor de Cumpărături . . . . .	33
4.2.7 Pagina pentru Reciclare . . . . .	34
<b>5 Concluzii</b>	<b>36</b>
<b>Bibliografie</b>	<b>38</b>

# Capitolul 1

## Introducere

### 1.1 Context

Principala autoritate globală din domeniul mediului, Programul Națiunilor Unite pentru Mediu (UNEP), a lansat anul acesta al treilea raport care furnizează date privind risipa alimentară. Conform acestui raport, în medie, fiecare persoană aruncă anual 79 de kg de alimente, iar pierderile și deșeurile alimentare sunt responsabile pentru aproape 10% din emisiile anuale de gaze cu efect de seră (GHG). [19] Cantitatea acestor gaze cu efect de seră (dioxid de carbon, metan, dioxid de azot) a crescut semnificativ în ultimele decenii din cauza activităților industriale, aceste gaze fiind responsabile pentru criza climatică.

Deoarece cei mai mari contribuitori la deșeurile alimentare nu sunt indivizii, ci marile corporații și supermarketuri [21], schimbările individuale nu sunt de ajuns, ci e nevoie de schimbări sistemice prin introducerea de politici noi și modificarea legislației.

Cu toate astea, conform ReFED [17], educarea consumatorilor ar putea preveni milioane de tone de emisii de GHG. Măsurile luate la nivel individual pot ajuta oamenii să își conștientizeze impactul asupra planetei, să fie în legătură cu lumea cu care trăiesc și, totodată, prin schimbarea obiceiurilor lor, ajung să pună presiune pe autorități, astfel să se producă o schimbare sistemică.

Tot ReFED menționează că, printre măsurile care pot fi luate de fiecare persoană sunt: planificarea unei liste de cumpărături sau folosirea creativității pentru a folosi alimentele rămase și reciclarea.

## **1.2 Motivație**

Aplicația “CookIt” a fost concepută inițial dintr-o motivație strict personală de a economisi bani, deoarece am observat că aruncarea alimentelor pe care nu reușesc să le folosesc la timp îmi afectează negativ bugetul personal.

În timpul cercetării aprofundate asupra acestei probleme, am realizat că risipa alimentară nu reprezintă doar o pierdere financiară la nivel individual, ci are consecințe negative mult mai ample, afectând întregul ecosistem și, aşa cum am menționat mai sus, este un factor contribuitor la criza climatică cu care ne confruntăm.

Risipa alimentară contribuie la degradarea mediului prin creșterea deșeurilor și emisiilor de gaze cu efect de seră, ceea ce m-a determinat să înțeleg importanța abordării acestei probleme dintr-o perspectivă globală, nu doar personală. Conștientizarea aceasta a fost un motiv de bază pentru dezvoltarea aplicației “CookIt”.

## **1.3 Scopul aplicației**

Aplicația mea își propune să ajute utilizatorii să gestioneze mai eficient risipa alimentară, să contribuie la reducerea ei și la protejarea mediului.

Oferă suport pentru multiple nevoi: reciclare, conștientizare, planificarea cumpărăturilor și identificarea variantelor de rețete utilizând ingredientele rămase. Aplicația se adresează celor care doresc să facă schimbări în stilul lor de viață, oferind acces la rețete simple.

Este destinată utilizatorilor care doresc să afle mai multe despre prevenirea risipei alimentare, să înceapă să recicleze, să facă mici schimbări în stilul de viață. Aplicația funcționează ca un instrument cu dublu rol, oferind sprijin pentru toate acestea și contribuind în același timp la o abordare mai sustenabilă și conștientă în gestionarea resurselor alimentare.

## **1.4 Aplicații similare**

Așa cum am menționat și mai sus, aplicația oferă mai multe funcționalități, dar cea mai importantă este cea care permite utilizatorului să găsească rețete cu orice combinație de ingrediente își dorește. Cele mai populare aplicații care oferă aceeași funcționalitate sunt: ”SuperCook” și ”MyFridgeFood”.

Funcționalități pe care "CookIt" nu le are:

1. **Lipsa unei baze de date extinse cu rețete predefinite:** SuperCook și MyFridgeFood beneficiază de o baza de date extinsă cu rețete predefinite, ceea ce permite o căutare rapidă. Aplicația mea generează rețete în timp real folosind ChatGPT, ceea ce poate duce la rezultate variaibile, durează mai mult timp și necesită o conexiune activă la internet pentru a accesa API-ul.
2. **Lipsa informațiilor nutriționale ale rețetei** MyFridgeFood oferă utilizatorului, odată ajuns pe pagina rețetei, informații despre ingredientele sale, posibilitatea de a consulta informațiile nutriționale ale rețetei. Gramajul de proteine, grăsimi și zaharuri cât și numărul de calorii sunt disponibile pentru orice rețetă.

Funcționalități pe care "CookIt" le are încă plus:

1. **Generarea Imagineilor:** Aplicația mea folosește DALL-E 3 pentru a genera imagini atractive ale rețetelor. Utilizatorii pot vizualiza cum ar arăta rețeta finală, ceea ce îmbunătățește considerabil experiența utilizatorului și poate veni ca suport pentru acesta în procesul de gătire, deoarece i s-a oferit o idee creativă de prezentare a preparatului.
2. **Salvarea și gestionarea rețetelor:** Utilizatorii aplicației "CookIt" pot salva rețetele generate și le pot accesa ulterior din pagina "Saved Recipes". Funcționalitatea aceasta oferă utilizatorului o experiență mult mai bună navigând în aplicație și îl face să se reîntoarcă la ea, deoarece poate găsi preparate care i-au respectat toate cerințele și pe care va mai vrea să le mai prepare. Aceasta fiind, de asemenea, o componentă care poate promova aplicația.

# Capitolul 2

## Tehnologii utilizate

### 2.1 ASP.NET Core

.NET reprezintă o platformă destinată aplicațiilor, care include instrumente complexe de dezvoltare, limbaje de programare și biblioteci. ASP.NET Core completează platforma .NET cu seturi de instrumente și biblioteci axate pe dezvoltarea aplicațiilor web.

În ceea ce privește performanța și eficiența, ASP.NET Core utilizează un limbaj compilat, în cazul aplicației mele, am folosit C#. Compilatorul său optimizează codul pentru a asigura o execuție rapidă și o utilizare eficientă a resurselor. Câteva avantaje ale ASP.NET sunt securitatea și gestionarea memoriei, realizată prin intermediul colectorului de gunoi, prevenind scurgerile. [10]

Astfel, ASP.NET Core este o opțiune favorabilă din mai multe motive: performanță, securitate, eficiență în dezvoltarea aplicațiilor web, îmbunătățește platforma .NET cu funcționalități specifice dezvoltării web, autentificare integră și un management eficient al datelor și resurselor.

### 2.2 Bootstrap

Bootstrap este un framework front-end open-source, utilizat pe scară largă pentru dezvoltarea de interfețe web responsive și mobile-first. Bootstrap oferă un set de instrumente și componente predefinite care simplifică procesul de creare al aplicațiilor web. [1]

#### Ce ofera Bootstrap?

1. Sistemul de grilă este baza pentru design-ul responsive, fiind folosită o grilă de 12 coloane care se ajustează automat la dimensiunea ecranului.

2. Oferă o gamă largă de componente UI predefinite, cum ar fi butoane, formulare, carduri, alerte, badge-uri, nav-uri, navbar-uri etc.
3. Include diverse plugin-uri JavaScript care adaugă funcționalități interactive pentru componentele UI. Aceste plugin-uri sunt bazate pe jQuery și includ multiple funcționalități.
4. Vine cu un set complex de stiluri CSS, care permit dezvoltatorilor să creeze teme personalizate.
5. E proiectat să funcționeze în majoritatea browserelor moderne, asigurând o experiență remarcabilă pentru utilizatori, în ciuda browserului utilizat de aceștia.

În dezvoltarea aplicației mele, pentru a realiza frontend-ul, am folosit Bootstrap completat de limbajele CSS, HTML și JavaScript.

CSS (Cascading Style Sheets) a fost utilizat pentru a stiliza elementele HTML și pentru a controla aspectul vizual al aplicației.

HTML (HyperText Markup Language) a fost folosit pentru structurarea conținutului paginilor web. Prin intermediul tag-urilor HTML, am definit scheletul paginilor, inclusiv diverse componente Bootstrap, cum ar fi grile, formulare, butoane și navegări, pentru a crea fluxul logic al aplicației.

JavaScript a fost folosit pentru a adăuga interactivitate și funcționalitate dinamică în aplicație. Am folosit biblioteca jQuery pentru a simplifica manipularea DOM și pentru a crea interacțiuni complexe între elementele paginii.

Prin combinarea acestor tehnologii, am reușit să dezvolt o aplicație web, care oferă o experiență pentru utilizatori cât mai plăcută.

## 2.3 ASP.NET Identity

ASP.NET Identity este un sistem de gestionare a identităților și autentificării utilizatorilor în cadrul aplicațiilor ASP.NET și oferă un framework flexibil și extensibil pentru autentificare și autorizare, permitând dezvoltatorilor să gestioneze utilizatorii, rolurile și alte informații legate de identitate și securitatea utilizatorului. Este proiectat pentru a înlocui sistemele anterioare de gestionare a utilizatorilor, cum ar fi Membership, Simple Membership și Universal Providers. [9]

**Ce oferă Identity Framework? [9]**

1. Permite utilizatorilor să se autentifice folosind parole, conturi de rețele sociale sau alți furnizori de identitate externi și, totodată, permite definirea și gestionarea rolurilor și a permisiunilor pentru utilizatori. Suportă autentificarea bazată pe cookie-uri, autentificarea prin tokenuri (JWT), autentificarea externă (OAuth, OpenID Connect) și autentificarea cu doi factori (2FA).
2. Utilizează Entity Framework pentru a stoca datele de identitate într-o bază de date. Poate fi configurat să utilizeze orice altă metodă de gestionare a datelor.
3. Permite dezvoltatorilor să extindă și să personalizeze schema de date și comportamentul implicit, pentru a se potrivi nevoilor specifice oricărei aplicații.
4. Oferă API-uri pentru gestionarea utilizatorilor, inclusiv înregistrare, resetarea parolei, modificarea parolei, blocare utilizatorilor, verificarea adresei de email și a numărului de telefon.

## 2.4 Entity Framework Core

Pentru stocarea datelor în cadrul unei aplicații ASP.NET MVC, se utilizează o tehnologie open source numită Entity Framework (EF). Entity Framework este un ORM (Object-Relational Mapper) pentru .NET, adică o colecție de biblioteci care asociază fiecare clasă dintr-un model cu o bază de date. Scopul utilizării EF este de a permite dezvoltatorilor să se concentreze pe dezvoltarea efectivă a aplicației, fără a se preocupa de complexitatea interacțiunii cu baza de date.

Deși procesarea datelor se poate realiza și prin metode tradiționale, cum ar fi utilizarea ADO.NET, EF oferă o modalitate mai eficientă de a implementa operațiunile CRUD (Create, Read, Update, Delete). În plus, Entity Framework permite utilizarea LINQ (Language Integrated Query), care facilitează integrarea oricărui RDBMS (Relational Database Management System) cum ar fi Oracle Database, SQL Server etc. [8]

Entity Framework Core (EF Core) este o versiune modernă a ORM-ului dezvoltată de Microsoft pentru .NET. EF Core permite dezvoltatorilor să interacționeze cu baza de date utilizând obiecte .NET, eliminând necesitatea de a scrie manual majoritatea codului SQL. [8] EF Core este proiectat să fie extensibil, performant și ușor de utilizat, oferind o soluție robustă pentru gestionarea datelor în aplicațiile .NET.

### Avantaje ale utilizării Entity Framework Core

1. Entity Framework Core oferă un nivel ridicat de abstracție, permitând dezvoltatorilor să interacționeze cu baza de date utilizând clase și obiecte. Aceasta simplifică semnificativ procesul de dezvoltare și întreținere a codului, deoarece operațiunile de acces

la date sunt efectuate prin intermediul unor entități definite în cod. În loc să scrie cod SQL direct, dezvoltatorii pot lucra cu obiecte .NET, ceea ce face ca aplicațiile să fie mai ușor de citit, de întreținut și de extins.

2. Entity Framework Core este compatibil cu o gamă largă de sisteme de gestionare a bazelor de date, inclusiv SQL Server, SQLite, MySQL și PostgreSQL. Această compatibilitate largă oferă flexibilitate și portabilitate aplicațiilor, permitându-le să funcționeze pe diverse platforme și să se adapteze la diferite medii de dezvoltare și producție. Dezvoltatorii pot schimba baza de date utilizată fără a fi necesară rescrierea semnificativă a codului de acces la date, datorită abstracției oferite de EF Core.

3. Funcționalitatea de migrare a Entity Framework Core permite dezvoltatorilor să schimbe schema bazei de date într-un mod controlat, fără a pierde datele existente. Migrations gestionează modificările schemei bazei de date, asigurând că baza de date rămâne sincronizată cu modelul aplicației. Acest proces facilitează actualizările și modificările structurale ale bazei de date, reducând riscul de erori și permitând o gestionare mai eficientă a evoluției bazei de date pe parcursul ciclului de viață al aplicației.

## 2.5 SQL Server

Baza de date are un rol foarte important în cadrul unei aplicații web, deoarece cu ajutorul ei se realizează stocarea datelor, manipularea datelor, securitatea și integritatea.

SQL Server este un sistem de gestionare a bazelor de date relaționale (RDBMS) dezvoltat de Microsoft. Este proiectat pentru a stoca și gestiona datele într-un mod structurat și accesibil, utilizând limbajul SQL (Structured Query Language). ASP.NET Framework oferă suport nativ pentru SQL Server prin diverse biblioteci și framework-uri, cum ar fi ADO.NET și Entity Framework. [12]

Am ales SQL Server deoarece, fiind dezvoltat de Microsoft, funcționează perfect cu .NET, Visual Studio, simplificând dezvoltarea și gestionarea aplicațiilor. Oferă indexare eficientă și capacitate de caching, îmbunătățind astfel viteza și eficiența accesului la date. În plus, SQL Server oferă funcționalități avansate de securitate, cum ar fi criptarea datelor, autentificare robustă și control al accesului bazat pe roluri.

## 2.6 ChatGPT API

Inteligenta artificială (IA) poate fi privită ca o unealtă contemporană care ne ajută să cunoaștem și să dezvoltăm lumea. E destul de greu de definit ce e exact IA, dar Tesler

spune că „Inteligenta artificială este tot ceea ce nu a fost încă făcut” [7] De exemplu, capacitatea unui algoritm de recunoaștere facială e o tehnologie des întâlnită în prezent, deci nu poate fi numită inteligență artificială, chiar dacă acum 50 de ani era. În acest spațiu în care algoritmii evoluează rapid, cea mai populară formă de IA prezentă în viețile noastre în ultimul an a fost ChatGPT.

ChatGPT (Generative Pre-trained Transformer) este un chatbot dezvoltat de OpenAI, bazat pe arhitectura Transformer. Acest model a fost lansat ca prototip în noiembrie 2022 sub versiunea GPT-3.5. În prezent, cea mai nouă versiune este GPT-4o, care e accesibilă doar printr-un abonament la OpenAI. ChatGPT este antrenat pentru a genera conținut care să simuleze cât mai mult manifestarea umană.

Arhitectura Transformer a fost introdusă pentru prima dată în 2017 în lucrarea ”Attention is All You Need” publicată de șapte cercetători de la Google. Această lucrare e considerată baza inteligenței artificiale aşa cum o percepem noi în prezent. [20] Transformer se bazează pe mecanisme de atenție, eliminând utilizarea rețelelor recurente (RNN) și convoluționale (CNN) și are cea mai mare performanță în îndeplinirea sarcinilor de procesare a limbajului natural (NLP).

Pe lângă mecanismele de atenție care oferă capacitatea modelului de a se concentra pe părțile relevante ale inputului la fiecare pas al generării textului, Transformer are și componenta de encoders și decoders. Encoders procesează inputul și creează reprezentări interne, iar decoders folosesc aceste reprezentări pentru a genera outputul efectiv.

Mecanismul de atenție funcționează după două principii: self-attention (atenția la sine) și multi-head attention. Fiecare cuvânt dintr-o propoziție are un sens doar în context cu restul cuvintelor, nu putem spune exact ce sens are cuvântul “viață” de exemplu, decât dacă îl putem analiza în contextul unei propoziții întregi. Așa că, prin mecanismul self-attention, fiecare cuvânt reține celealte cuvinte și sensul lor, în raport unele cu celealte. Multi-head attention se referă la faptul că, în paralel, mai multe capete preiau sensul unui text din perspective diferite. De exemplu, un cap poate fi concentrat pe relațiile de subiecte și predicate, în timp ce alt cap poate analiza relațiile de adjective și substantive.

GPT folosește doar partea de decodificare a arhitecturii Transformer. Este pre-antrenat pe o cantitate uriașă de date text dintr-o plajă de surse multiple (date web crawl, wikipedia, literatură, articole de știri etc) [sursa: documentatia de la chat]. Este antrenat să prezică următorul cuvânt într-o secvență, folosind un set de date textuale mari, urmând a fi ajustat ulterior pentru sarcini specifice (fine-tuning). [14]

Chat GPT API permite dezvoltatorilor să comunice cu modelul GPT pentru a genera răspunsuri în funcție de promptul dat de utilizator. Acest proces se face prin trimite-reia unor cereri HTTP și primirea răspunsurilor în format JSON. Pentru a folosi API-ul, dezvoltatorii trebuie să se autentifice, să trimită cererile corespunzătoare, să proceseze răspunsurile primite și să gestioneze eventualele erori. [14]

DALL-E 1, prima versiune a modelului, a demonstrat capacitatea de a genera imagini coerente și creative, însă avea limitări în ceea ce privește claritatea și complexitatea detaliilor. [13] Versiunile ulterioare, DALL-E 2 și 3, au îmbunătățit semnificativ calitatea imaginilor, adăugând detalii mai multe și o înțelegere mult mai bună a relațiilor spațiale și contextuale între obiecte.

Cel mai nou model dezvoltat de OpenAI este Sora, model antrenat să genereze videoclipuri pe baza unor descrieri sub formă de text.

În realizarea aplicației mele, am utilizat două dintre API-urile furnizate de OpenAI: Chat GPT API pentru generarea de text și DALL-E API pentru generarea de imagini. Aceste tehnologii se bazează pe modele de învățare care sunt antrenate pe seturi mari de date pentru a oferi capacitați avansate de procesare a limbajului natural și generare de imagini.

```
1 def get_completion(ingredients, model="gpt-3.5-turbo"):
2     prompt = f"Generate a recipe in English, including the preparation
3             steps and a complete list of ingredients, using the following all of
4             the ingredients: "
5     prompt += ", ".join(ingredients)
6     messages = [{"role": "user", "content": prompt}]
7     response = openai.ChatCompletion.create(
8         model=model,
9         messages=messages,
10        temperature=0,
11    )
12
13     return response.choices[0].message["content"]
```

Listing 2.1: Funcția care accesează API-ul ChatGPT

În codul pentru generarea de rețete cu ChatGPT API, cheia API este setată pentru autentificare folosind openai.api\_key. Funcția get\_completion construiește promptul, care solicită modelului GPT-3.5-turbo să genereze o rețetă bazată pe ingredientele furnizate de utilizator. Promptul este formulat pentru a include toate ingredientele date, iar răspunsul returnat de API conține o rețetă detaliată, cu lista de ingrediente și pașii de preparare. [16]

```

1 def generate_image(recipe):
2     prompt = f"Generate an image of this freshly prepared dish, making it
3         as appetizing as possible: {recipe}"
4     response = openai.Image.create(
5         model="dall-e-3",
6         prompt=prompt,
7         n=1,
8         size="1024x1024"
9     )
10    return response['data'][0]['url']

```

Listing 2.2: Funcția care accesează API-ul DALL-E

În codul pentru generarea de imagini cu DALL-E API, funcția ‘generate\_image’ construiește un prompt care descrie vizual preparatul culinar generat anterior și apelează API-ul DALL-E 3. Promptul este titlul rețetei, preluat din textul generat de ChatGPT, și oferă detalii despre cum ar trebui să arate preparatul, cerând modelului să creeze o imagine corespunzătoare. Răspunsul API-ului returnează un URL al imaginii generate. [15]

Eu am ales să folosesc API-ul ChatGPT datorită cantității de date pe care e antrenat și al performanței sale. Cred cu tărie că aplicațiile bune sunt cele care reușesc să facă ceva nou din ceva ce deja există.

## 2.7 YouTube Data API

YouTube Data API oferă dezvoltatorilor acces la funcționalitățile și datele platformei YouTube, cu mai multe limitări care țin de politica platformei, la momentul actual având protocolul OAuth 2.0 pentru autorizarea accesului la datele private ale utilizatorilor. API-ul permite integrarea capacităților YouTube în aplicații web și mobile, facilitând interacțiunea cu resursele YouTube, cum ar fi videoclipuri, playlist-uri sau canale.[5]

### Funcționalități oferite de YouTube Data API:

1. Search Result: această funcționalitate furnizează informații despre un videoclip YouTube care corespunde parametrilor de căutare specificați într-o cerere API. Deși rezultatul unei căutări indică o resursă unică identificabilă, cum ar fi un videoclip, acesta nu deține date persistente proprii.

Am utilizat funcționalitatea de căutare pentru a obține informații relevante despre videoclipuri educative privind prevenirea risipei alimentare. Conform promptului de că-

utare specific, aplicația returnează o listă de rezultate relevante, fiecare conținând detalii precum titlul videoclipului, descrierea și link-ul către videoclip.

2. Thumbnail: această funcționalitate identifică imaginile asociate cu o resursă, în acest caz un videoclip YouTube.

În aplicația mea, am folosit funcționalitatea de thumbnail pentru a afișa imaginile asociate ale videoclipurilor returnate din rezultatele căutării despre care am menționat anterior. Astfel, utilizatorii pot vedea acces la o previzualizare a fiecărui videoclip înainte de a face click pe link, pentru a viziona conținutul întreg pe YouTube.

3. Video: reprezintă un singur videoclip YouTube, incluzând toate informațiile relevante despre acel videoclip.

Am folosit funcționalitatea video pentru a prelua informații detaliate despre videoclipurile individuale. Aceste detalii sunt prezentate utilizatorilor în aplicație pentru a le oferi o imagine completă a conținutului înainte de vizionare.

Pașii utilizării API-ului YouTube Data:

### 1. Configurarea

În prima etapă a integrării YouTube Data API în aplicație, am configurat parametrii esențiali pentru efectuarea cererilor API. Această configurare a inclus definirea cheii API (apiKey), care este necesară pentru autentificarea cererilor către YouTube Data API. Am stabilit promptul de căutare (searchTerm), de exemplu "food waste prevention", pentru a specifica ce videoclipuri trebuie să fie căutate. De asemenea, am setat numărul maxim de rezultate returnate (maxResults), pentru a limita și organiza mai bine informațiile afișate utilizatorilor. Toate aceste variabile au fost folosite pentru a construi URL-ul cererii API, care a fost format pentru a interoga YouTube Data API și a returna videoclipuri pe baza termenului de căutare specificat.

### 2. Descărcarea răspunsurilor API

După configurarea parametrilor, am folosit clasa System.Net.WebClient pentru a trimite cererea către YouTube Data API și pentru a descărca răspunsul. Acest răspuns, primit în format JSON, a fost deserializat folosind biblioteca Newtonsoft.Json.JsonConvert, care mi-a permis transformarea datelor JSON într-un format structurat și ușor de manipulat în cadrul aplicației. Această etapă a asigurat obținerea informațiilor relevante despre videoclipuri, cum ar fi ID-ul videoclipului, titlul și thumbnail-ul, permitând procesarea și afișarea acestor date utilizatorilor.

### 3. Procesarea răspunsurilor și afișare video

În etapa finală, am procesat răspunsul API și am afișat videoclipurile în aplicație.

Am parcurs fiecare element din data.items pentru a obține videoId, videoTitle și thumbnailUrl. Aceste informații au fost folosite pentru a genera HTML care afișează fiecare videoclip într-un format de thumbnail, inclusiv un link către videoclipul YouTube și titlul videoclipului. Funcția ExtractYouTubeVideoID a fost utilizată pentru a extrage ID-ul videoclipului dintr-un URL complet de YouTube, iar funcția GetYouTubeVideoTitle a obținut titlul unui videoclip YouTube folosind ID-ul videoclipului și API-ul YouTube. În urma acestui proces, utilizatorii pot să vizualizeze și să acceseze ușor videoclipurile relevante direct din aplicație.

## 2.8 Google Maps API

Google Maps API este un set de interfețe de programare a aplicațiilor, furnizat de Google, care permite dezvoltatorilor să integreze hărți și funcționalități de localizare în aplicațiile lor web sau mobile. API-urile Google Maps oferă o gamă largă de servicii și instrumente care permit crearea de aplicații personalizate, interactive și bazate pe localizare.

### Maps JavaScript API: [4]

- Permite integrarea unei hărți interactive în paginile web. Dezvoltatorii pot personaliza hartă, adăuga marcaje (markers), ferestre de informații (info windows), și straturi (layers) suplimentare.
- Suportă diverse controale pentru navigație, cum ar fi zoom-ul și schimbarea tipului de hartă (satelit, teren, hibrid).

Am utilizat în aplicația mea Google Maps API pentru a afișa o hartă interactivă pe o pagină web. Funcțiile definite permit obținerea locației utilizatorului și afișarea locațiilor de reciclare pentru diferite materiale, cum ar fi hârtia, plasticul, sticla și metalul. De asemenea, sunt incluse funcții pentru ștergerea markerelor existente de pe hartă. Interactivitatea este îmbunătățită prin utilizarea jQuery și Bootstrap pentru gestionarea evenimentelor și stilizarea elementelor UI.

Funcția getUserLocation utilizează API-ul de geolocalizare HTML5 pentru a obține locația curentă a utilizatorului. Dacă geolocalizarea este disponibilă și permisă de utilizator, coordonatele sunt utilizate pentru a centra harta și a plasa un marker la locația utilizatorului.

```
1 function getUserLocation() {  
2     if (navigator.geolocation) {  
3         navigator.geolocation.getCurrentPosition(function (position) {
```

```

4      var userLocation = {
5          lat: position.coords.latitude,
6          lng: position.coords.longitude
7      };
8      map.setCenter(userLocation);
9      var userMarker = new google.maps.Marker({
10          position: userLocation,
11          map: map,
12          title: 'My Location'
13      });
14      function (error) {
15          console.error("Eroare la obtinerea locatiei:", error);
16      });
17 } else {
18     console.error("Geolocation nu este suportat de browser-ul
19         utilizatorului");
20 }

```

Listing 2.3: Funcția care preia locația utilizatorului

Funcția getPaperLocation curăță marker-ele anterioare pentru hârtie, definește noi locații pentru reciclarea hârtiei și plasează marker-uri pe hartă pentru aceste locații. Marker-ele sunt stocate în array-ul paperMarkers. Funcții similare sunt definite pentru locațiile de reciclare a plasticului, sticlei și metalului (getPlasticLocation, getGlassLocation, getMetalLocation).

Evenimente de click adaugă evenimente de click pentru butoanele "My Location", "Paper", "Plastic", "Glass" și "Metal". Acestea apeleză funcțiile corespunzătoare pentru a obține locațiile și a afișa marker-ele pe hartă sau pentru a curăța marker-ele existente. Google Maps API include scriptul Google Maps API cu cheia API specificată și apeleză funcția initMap pentru inițializarea hărții atunci când scriptul este încărcat.

# Capitolul 3

## Implementarea aplicației

### 3.1 Baza de date

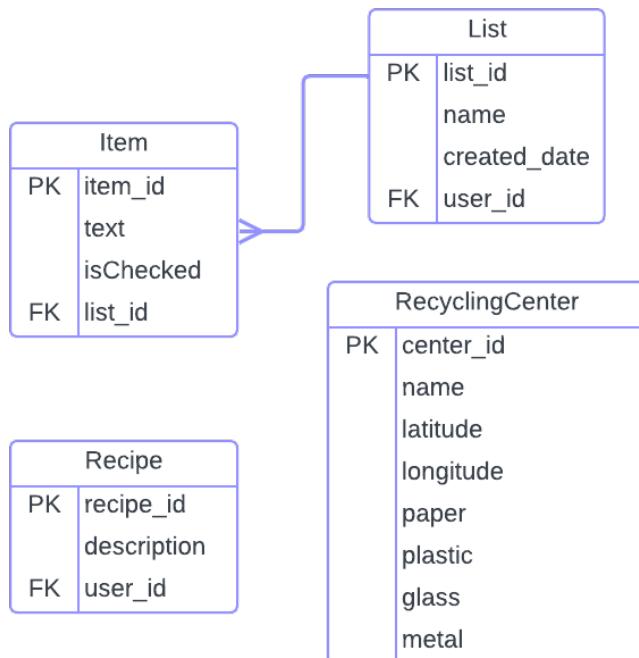


Figura 3.1: Diagrama entitate-relație

Contextul de date gestionează conexiunile și operațiunile cu baza de date, facilitând interacțiunea aplicației cu baza de date SQL Server. Entitățile definite în cadrul acestui context sunt utilizate de controlere pentru a efectua operațiuni CRUD (Create, Read, Update, Delete). Acest proces asigură o interacțiune eficientă și sigură între aplicație și baza de date, permitând gestionarea corespunzătoare a datelor și menținerea integrității acestora.

Pentru a conecta aplicația web la baza de date, am urmat pașii de mai jos: [11]

1. În SQL Server Object Explorer se deschide secțiunea (localdb)\MSSQLLOCALDB și se adaugă o nouă bază de date.
2. În Solution Explorer, la secțiunea Connected Services, se adaugă SQL Server Database, SQL Server Express LocalDB. Aici se adaugă numele bazei de date. De aici se salvează și stringul de conectare, care e dat ca parametru în metoda de mai sus și este unic în cadrul fiecărei baze de date.
3. Ultimul pas e cel în care se adaugă stringul de conectare la baza de date ca parametru al metodei UseSqlServer() în appsetting.json, pentru variabila DefaultConnection.

După conectarea la baza de date, se adaugă o migrație “Add-Migration nume\_migratie” și se actualizează baza de date folosind comanda “Update-Database”. [18]

## 3.2 Model-View-Controller

Model-View-Controller (MVC) reprezintă un model arhitectural utilizat în dezvoltarea aplicațiilor. Cu ajutorul acestei arhitecturi se separă logica aplicației de interfața cu utilizatorul.

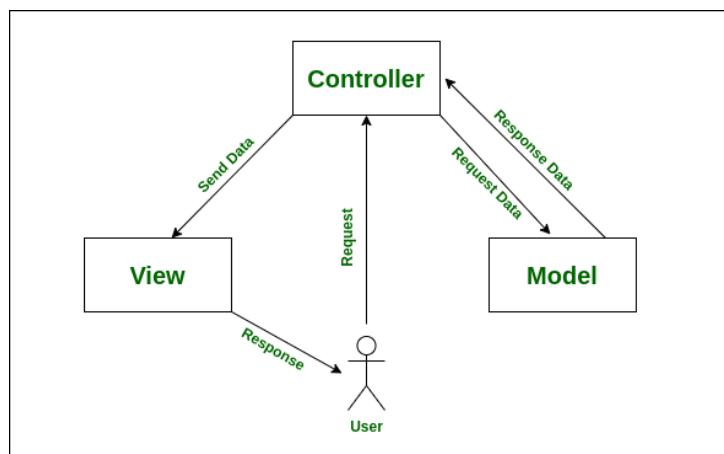


Figura 3.2: Reprezentare a arhitecturii MVC [3]

Această separare rezultă într-o aplicație în care aspectul vizual (frontend-ul) și backend-ul pot fi modificate mai ușor, fără a afecta celelalte componente ale sistemului. Această izolare permite ca întreținerea și dezvoltarea ulterioară a aplicației să fie făcută cu o flexibilitate mai mare.

### 3.2.1 Models

Modelul este responsabil pentru gestionarea și manipularea datelor din aplicație. Aceasta răspunde la cererile provenite din View prin intermediul Controller-ului. Modelul se ocupă de procesarea și manipularea datelor și e cel mai jos nivel al arhitecturii MVC, constituind nucleul aplicației. El este responsabil pentru realizarea conexiunii cu baza de date și pentru asigurarea integrității și consistenței datelor gestionate de aplicație.

Clasa List se ocupă cu gestionarea listelor de cumpăraturi create de utilizatori. Fiecare utilizator care își creează un cont în aplicație are posibilitatea de a-și crea sau să stearge liste de cumpăraturi în pagina “shopping lists”.

- Id: Reprezintă codul unic al listei. Este de tip int și este folosit pentru a diferenția fiecare listă în baza de date.
- Name: Reprezintă numele listei. Este de tip string și poate fi null. Acest câmp poate stoca numele sau titlul listei.
- UserId: Reprezintă identificatorul utilizatorului care a creat lista. Este de tip string și este folosit pentru a asocia lista cu un utilizator specific.
- CreatedDate: Reprezintă data la care a fost creată lista. Este de tip DateTime și adaugă informații despre momentul în care lista a fost generată. Adăugarea acestui câmp permite păstrarea evidenței privind momentul creării fiecărei liste, ceea ce poate fi util pentru organizarea și filtrarea datelor. Am ales să am și acest câmp, deoarece asta permite listelor să fie afișate în pagină în ordinea descrescătoare creării, astfel utilizatorul poate vedea prima data cele mai recente liste create, ceea ce face experiența utilizării mai bună.
- Item: Reprezintă colecția de elemente (Item) din listă. Este de tip ICollection<Item> și este inițializat cu o listă goală de Item. Aceasta permite gestionarea multiplelor elemente asociate cu lista. Inițializarea colecției în constructor (new List<Item>()) asigură că Items nu va fi niciodată null, evitând astfel potențialele erori la manipularea listelor.

Clasa Recipe are scopul de a salva rețetele generate pe care utilizatorii logați vor să le pastreze.

- Id: Reprezintă codul unic al rețetei. Este de tip int și este folosit pentru a diferenția fiecare rețetă în baza de date.
- Description: Reprezintă textul care a fost generat anterior de ChatGPT. Este de tip string și poate fi null (optional). Acest câmp poate conține informații suplimentare despre rețetă, cum ar fi pașii de preparare, ingrediente, etc.

- UserId: Reprezintă identificatorul utilizatorului care a salvat rețeta. Este de tip string și este folosit pentru a asocia rețeta cu un utilizator specific.

Clasa Recycling Center reține date despre coordonatele geografice ale centrelor de reciclare, pentru a putea fi afișate pe harta preluată cu API-ul de la Google Maps.

- Id: Reprezintă codul unic al centrului de reciclare. Este de tip int și este folosit pentru a diferenția fiecare centru în baza de date.
- Name: Reprezintă numele centrului de reciclare. Este de tip string și stochează denumirea centrului.
- Latitude: Reprezintă latitudinea centrului de reciclare. Este de tip double și este folosit pentru a stoca coordonata geografică nord-sud a locației centrului.
- Longitude: Reprezintă longitudinea centrului de reciclare. Este de tip double și este folosit pentru a stoca coordonata geografică est-vest a locației centrului.
- Proprietățile Paper, Plastic, Metal și Glass indică tipurile de materiale pe care centrul de reciclare le acceptă. Acestea sunt utile pentru utilizatorii care doresc să știe unde pot recicla anumite tipuri de deșeuri.

### 3.2.2 Views

View-ul reprezintă interfața cu utilizatorul și este componenta arhitecturii MVC cu care utilizatorii interacționează prin intermediul browser-ului, mai pe scurt, prin intermediul view-urilor se realizează frontend-ul unei aplicații web și acolo sunt afișate toate datele. În cadrul arhitecturii MVC, View-ul comunică direct doar cu Controller-ul, iar interacțiunea cu Modelul se face tot prin intermediul Controller-ului.

GenerateView: reprezintă o interfață de utilizator pentru generarea rețetelor pe baza ingredientelor furnizate. Utilizatorii pot introduce ingredientele pe care le au și pot adăuga câmpuri suplimentare pentru mai multe ingrediente. Formularul validează că sunt completeate cel puțin trei câmpuri înainte de trimitere. Stilizarea paginii este personalizată pentru a crea o experiență vizuală plăcută, iar un script JavaScript gestionează validarea și adăugarea dinamică a câmpurilor de ingrediente.

GeneratedRecipe: reprezintă interfața de utilizator pentru afișarea unei rețete generate și a unei imagini asociate, dacă este disponibilă. Utilizatorii pot vedea rezultatul generării rețetei și au opțiunea de a salva rețeta printr-un formular. Stilizarea paginii este personalizată pentru a crea o experiență vizuală plăcută, iar structura conținutului este

organizată eficient pentru a separa imaginea de textul rețetei.

Home: reprezintă interfața de utilizator pentru pagina de start a aplicației, oferind un fundal atractiv și un buton proeminent care invită utilizatorii să genereze rețete. Butonul este stilizat pentru a atrage atenția, iar textul său oferă indicații clare despre funcționalitatea disponibilă. Spațiul suplimentar adăugat permite utilizatorilor să facă scroll, oferind o experiență de navigare fluidă.

List: permite utilizatorilor să gestioneze listele lor de cumpărături, inclusiv adăugarea, ștergerea și modificarea elementelor din liste. Stilurile CSS și scripturile JavaScript asigură o interfață vizuală plăcută și funcțională. Utilizatorii pot adăuga noi liste, bifă elementele existente ca fiind complete, și pot șterge elementele sau listele întregi.

Recile: este responsabil pentru afișarea unei pagini de reciclare unde utilizatorii pot vizualiza și selecta centrele de reciclare pentru diverse materiale, cum ar fi hârtie, plastic, sticlă și metal. Pagina include un fundal stilizat, butoane pentru selectarea materialelor de reciclat și o hartă interactivă pentru afișarea locațiilor centrelor de reciclare.

Recipe: afișează o listă de rețete salvate de utilizator. Folosește o imagine de fundal pentru a îmbunătăți aspectul vizual și include CSS personalizat pentru stilizare. Fiecare rețetă este prezentată într-o secțiune delimitată, stilizată cu un font specific pentru a îmbunătăți lizibilitatea și consistența estetică. Dacă nu există rețete salvate, se afișează un mesaj care încurajează utilizatorul să găsească și să salveze rețetele preferate. Pagina este proiectată pentru a fi atât funcțională, cât și plăcută vizual, facilitând navigarea utilizatorilor prin rețetele lor salvate.

Video: afișează o colecție de videoclipuri YouTube legate de prevenirea risipei alimentare, oferind utilizatorilor informații actualizate despre un stil de viață sustenabil. Pagina include un fundal stilizat și utilizează CSS pentru a aranja videoclipurile într-un format de tip galerie. Fiecare videoclip este prezentat cu un thumbnail și un titlu, iar utilizatorii pot da clic pe thumbnail pentru a viziona videoclipul pe YouTube. În cazul în care nu sunt găsite videoclipuri, se afișează un mesaj corespunzător. Funcțiile integrate permit obținerea și afișarea titlurilor și thumbnail-urilor videoclipurilor direct de pe YouTube folosind API-ul YouTube Data v3.

### 3.2.3 Controllers

Controller-ul este reprezentat de clase și constituie componenta care gestionează accesul la aplicație. În cadrul Controller-ului, se desfășoară următoarele procese: procesarea

cererilor HTTP, citirea datelor introduse de utilizator, trimitera datelor către Model pentru efectuarea operațiunilor necesare, transmiterea răspunsului către View. Controllerele din aplicația mea utilizează Entity Framework pentru interacțiunea cu baza de date și ASP.NET Identity pentru gestionarea utilizatorilor.

Metodele controllerul Generate:

- Generate: primește o listă de ingrediente, apelează un script Python pentru a genera o rețetă și o imagine corespunzătoare, și returnează vizualizarea "GeneratedRecipe" cu rețeta și imaginea genera.
- Save: salvează o rețetă în baza de date asociată utilizatorului curent autentificat și redirecționează către acțiunea "Index" din controller-ul "Recipe".
- Favorite: recuperează rețetele salvate din sesiune și le afișează în vizualizarea "Index" din controller-ul "Recipe".
- GetCompletionFromPythonScript: este responsabilă pentru a apela un script Python extern care generează o rețetă bazată pe ingredientele furnizate de utilizator. Aceasta funcție utilizează procesul de start pentru a executa scriptul Python și pentru a capta rezultatul generat.
- GetImageFromPythonScript: apelează un script Python pentru a genera o imagine pe baza unui prompt și returnează rezultatul ca string.

Metodele controllerul Home:

- Index: aceasta acțiune returnează vizualizarea implicită pentru pagina de start a aplicației și returnează un obiect de tip IActionResult, care este folosit pentru a reprezenta rezultatul unei acțiuni. La final se returnează vizualizarea asociată acțiunii Index.
- Privacy: returnează vizualizarea pentru pagina de confidențialitate și returnează un obiect de tip IActionResult. [Authorize] este un atribut care indică faptul că numai utilizatorii autentificați pot accesa această acțiune. Dacă un utilizator neautentificat încearcă să acceseze această acțiune, acesta va fi redirecționat către pagina de autentificare.
- Error: gestionază erorile și returnează vizualizarea pentru pagina de erori.

Acest controller folosește un logger pentru a înregistra informații și utilizează atrbute pentru a controla comportamentul acțiunilor. De exemplu, ILogger<HomeController>

injectează un obiect de tip ILogger pentru a înregistra informații legate de execuția controllerului.

Metodele controllerului List:

- Index: returnează vizualizarea principală care afișează toate listele utilizatorului curent autentificat.
- AddList: adaugă o nouă listă în baza de date și răspunde la cererile HTTP POST. Actiunea obține ID-ul utilizatorului curent și creează o nouă listă cu numele furnizat, asociată ID-ului preluat și setează data creării. Adaugă lista în contextul bazei de date și salvează modificările.
- AddItem: adaugă un nou element într-o listă existentă și răspunde la cererile HTTP POST. Se găsește lista corespunzătoare listId și, dacă lista există, creează un nou element cu textul furnizat și îl adaugă în listă apoi salvează modificările în contextul bazei de date.
- ToggleItem: comută starea IsChecked a unui element (marchează/dezmarchează). Se găsește elementul corespunzător ID-ului și, dacă elementul există, comută starea IsChecked și salvează modificările în contextul bazei de date.
- DeleteItem: șterge un element dintr-o listă. Găsește elementul corespunzător ID-ului și, dacă elementul există, îl șterge și salvează modificările în contextul bazei de date.
- DeleteList: șterge o listă și toate elementele asociate acesteia. Se găsește lista corespunzătoare ID-ului, inclusiv elementele asociate acesteia. Dacă lista există, șterge toate elementele asociate și apoi lista din contextul bazei de date și salvează modificările.

Metodele controllerului Recipe:

- Index: returnează vizualizarea principală care afișează toate rețetele utilizatorului curent autentificat. Atributul [Authorize] asigură că doar utilizatorii autentificați pot accesa această acțiune. Actiunea obține ID-ul utilizatorului curent și interoghează baza de date pentru a obține rețetele acestuia.
- Delete: acțiunea găsește rețeta corespunzătoare ID-ului, șterge rețeta din contextul bazei de date și salvează modificările.

Metodele controllerului RecyclingCenter:

- GetRecyclingCenters: această acțiune răspunde la cererile HTTP GET. Returnează centrele de reciclare în funcție de categoria materialului care e primită ca parametru. Numele categoriei este modificat la litere mici pentru a asigura o comparație insensibilă la majuscule apoi se interoghează baza de date pentru a obține centrele de reciclare care corespund categoriei furnizate. Se transformă centrele de reciclare într-o listă de obiecte simplificate, care conțin doar numele, latitudinea și longitudinea. Returnarea listei de centre de reciclare sub formă de JSON.

## 3.3 Funcționalitățile aplicației

### 3.3.1 Autentificarea

Utilizatorii se pot înregistra printr-un formular care colectează informațiile personale, de exemplu email, parolă, nume de utilizator. Metoda Register din AccountController gestionează procesul de înregistrare a utilizatorilor. De asemenea, dacă utilizatorii au deja un cont, ei se pot autentifica printr-un formular de login unde furnizează email-ul și parola. Metoda Login din AccountController gestionează procesul de autentificare a utilizatorilor.

Dacă autentificarea reușește, ASP.NET Identity generează un cookie de autentificare care este trimis clientului. Cookie-ul conține un token de autentificare criptat, care este utilizat pentru a identifica utilizatorul la fiecare cerere ulterioară, unde middleware-ul de autentificare verifică prezența cookie-ului de autentificare și mai apoi valizează procesul de autentificare.

### 3.3.2 Generarea Rețetelor

Utilizatorul accesează pagina Generate unde este întâmpinat cu un formular ce permite introducerea ingredientelor. Formularul este conceput pentru a accepta multiple intrări de text, corespunzătoare fiecarui ingredient. Acțiunea Generate din controller returnează vizualizarea Generate, unde utilizatorul poate introduce ingredientele dorite.

După ce utilizatorul completează formularul și trimită datele, acestea sunt preluate de către controller și transmise către scriptul de Python care utilizează API-ul ChatGPT pentru a genera rețeta. În această metodă, lista de ingrediente este preluată și transmisă funcției GetCompletionFromPythonScript. Rezultatul generat este rețeta sub formă de

text.

Pe baza titlului rețetei generate de ChatGPT, se apelează un alt script Python care utilizează modelul DALL-E 3 pentru a genera o imagine corespunzătoare aceluia text. Această imagine este apoi afișată alături de rețetă. Funcția GetImageFromPythonScript apelează scriptul Python care utilizează API-ul DALL-E 3 pentru a genera o imagine pe baza promptului de imagine derivat din titlul rețetei.

Deci, mai pe scurt, implementarea funcționalității de generare a rețetelor implică mai multe etape: preluarea ingredientelor de la utilizator, generarea rețetei folosind API-ul ChatGPT, generarea unei imagini relevante folosind API-ul DALL-E 3 și afișarea rezultatelor într-o pagină web.

### 3.3.3 Salvarea Rețetelor

Funcționalitatea de salvare a rețetelor permite utilizatorilor să păstreze rețetele generate pentru referințe ulterioare. Această funcționalitate asigură o experiență personalizată pentru fiecare utilizator, utilizând ID-ul acestora pentru a gestiona procesul de salvare rețetele.

În pagina Generated Recipe, utilizatorul vede un buton "Save Recipe" care permite salvarea rețetei generate. Acest buton inițiază o cerere POST către server pentru a salva rețeta în baza de date. Când utilizatorul apasă butonul "Save Recipe", datele sunt trimise către server. Acestea sunt gestionate de o metodă din controller care salvează rețeta în baza de date, asociind-o cu ID-ul utilizatorului curent.

### 3.3.4 Gestionarea Listei de Cumpărături

Această funcționalitate oferă utilizatorilor posibilitatea de a adăuga liste noi, de a sterge liste sau itemuri din listele existente și de a bifă alimentele cumpărate utilizând un checkbox. Utilizatorii care sunt autentificați în aplicație pot accesa o pagină unde pot vizualiza toate listele lor de cumpărături. În această pagină, ei pot adăuga liste noi și pot vedea toate listele create anterior. Metodele din controller gestionează adăugarea de liste și itemuri, ștergerea acestora și bifarea itemurilor cumpărate.

### 3.3.5 Accesare Centrelor de Reciclare

Funcționalitatea aceasta le permite utilizatorilor să interacționeze cu o hartă și să caute centrele de reciclare în funcție de materialele reciclabile: "hartie", "plastic", "sticla" sau "metal". Butoanele inițiază cereri către server pentru a obține locațiile centrelor de

reciclare și pentru a le afișa pe hartă. Utilizatorii pot selecta materialul dorit, iar harta, preluată prin API-ul Google Maps, afișează locațiile centrelor de reciclare relevante sub formă de pin-uri.

Când utilizatorul apasă pe unul dintre butoane, se trimit o cerere AJAX către server pentru a obține locațiile centrelor de reciclare corespunzătoare. Aceste locații sunt apoi afișate pe hartă sub formă de pin-uri. AJAX, acronimul pentru Asynchronous JavaScript and XML, permite actualizarea părților unei pagini web fără a fi necesară reîncărcarea întregii pagini. Acest lucru îmbunătățește experiența utilizatorului, făcând interacțiunea cu aplicația web mai rapidă și mai fluidă.

### **3.3.6 Vizualizarea Știrilor**

Funcționalitatea de integrare a YouTube API permite utilizatorilor aplicației noastre să fie la curent cu cele mai recente videoclipuri despre prevenirea risipei alimentare. Utilizatorii pot accesa aceste videoclipuri direct din pagina "News" a aplicației, oferindu-le o resursă valoroasă pentru a învăța și aplica metode eficiente de prevenire a risipei alimentare. Utilizarea YouTube Data API și jQuery asigură o experiență dinamică și interactivă, menținând utilizatorii informați.

# Capitolul 4

## Prezentarea aplicației

### 4.1 UI/UX

Scopul meu în dezvoltarea acestei aplicații a fost să mă axeze pe experiența utilizatorului atât vizuală cât și cea funcțională, astfel încât să îi îndeplinească nevoile și să se întoarcă la aplicație. Să fie o aplicație memorabilă la care să revii din necesitate și din dorință.

User Interface (UI) se concentrează pe experiența utilizatorilor cu interfața, să fie una cât mai plăcută și mai intuitivă. [2] User Experience (UX) se concentrează pe experiența generală a utilizatorilor cu aplicația și cu funcționalitățile sale, să răspundă nevoilor consumatorului. [6]

- Design Vizual: Am ales ca aplicația să se concentreze pe culoarea verde și alte culori pastelate, deoarece verdele conduce psihicul spre natură și e o culoare totodată asociată cu mâncarea. M-am axat pe contraste în layout și am folosit fontul 'Dancing Script', cursive.
- Elemente de Interfață: Butoanele sunt toate în aceeași tematică de culoare, intuitiv de folosit și de dimensiuni mari pentru a fi ușor observate. La fel și câmpurile pentru introducerea datelor, toate acestea fiind însoțite de placeholder-e care sunt intuitive.
- Navigație: Structura de navigație a site-ului e concentrată în meniul din navigation bar, care concentrează toate paginile accesibile utilizatorului într-un singur loc.
- Responsivitate: Am proiectat interfața ca să aibă capacitatea de a se adapta la diferite dimensiuni ale ecranului desktop.

UX se referă la experiența generală a utilizatorului atunci când interacționează cu aplicația. Aceasta include percepțiile, emoțiile și reacțiile utilizatorilor la interfața și funcționalitățile aplicației.

- Utilizare: Aplicația este intuitivă și ușor de utilizat, permite utilizatorilor să fie folosita, fiind în fiecare pagină însotită de toate indicațiile necesare.
- Performanță: Aplicația răspunde rapid la interacțiunile utilizatorilor și gestionează eficient cererile.
- Accesibilitate: Aplicația momentan nu este accesibilă pentru toți utilizatorii și nu este încă adaptată pentru persoane cu dizabilități.

## 4.2 Interfața

### 4.2.1 Pagina de Autentificare

Dacă un utilizator nu și-a creat încă un cont, acesta nu poate accesa opțiunile de salvare rețete și crearea unei liste de cumpărături. Dacă accesează aceste pagini, acesta e redirectionat către pagina de registrare. Dacă credențialele sunt valide, registrarea se realizează cu succes, iar în caz contrar este afișat un mesaj sugestiv.

The screenshot shows a registration form titled "Register as a new user". The fields are as follows:

- Email: cosmina.ducagmail.com (Error message: The Email field is not a valid e-mail address.)
- Password: \*\*\*\* (Error message: The Password must be at least 6 and at max 100 characters long.)
- Confirm Password: \*\*\*\*
- Username: Cosmina

A blue "Register" button is at the bottom, and a link "Already have an account? [Login here.](#)" is at the bottom right.

Figura 4.1: Procesul de înregistrare nevalidat

The screenshot shows a registration form titled "Register as a new user". The fields are as follows:

- Email: cosmina.duca@gmail.com
- Password: \*\*\*\*\*
- Confirm Password: \*\*\*\*\*
- Username: Cosmina

A blue "Register" button is at the bottom, and a link "Already have an account? [Login here.](#)" is at the bottom right.

Figura 4.2: Procesul de înregistrare validat

Dacă utilizatorul are deja un cont, acesta poate accesa pagina de autentificare. Dacă credențialele corespund unei intrări din baza de date, autentificarea se realizează cu succes, iar în caz contrar este afișat un mesaj sugestiv.

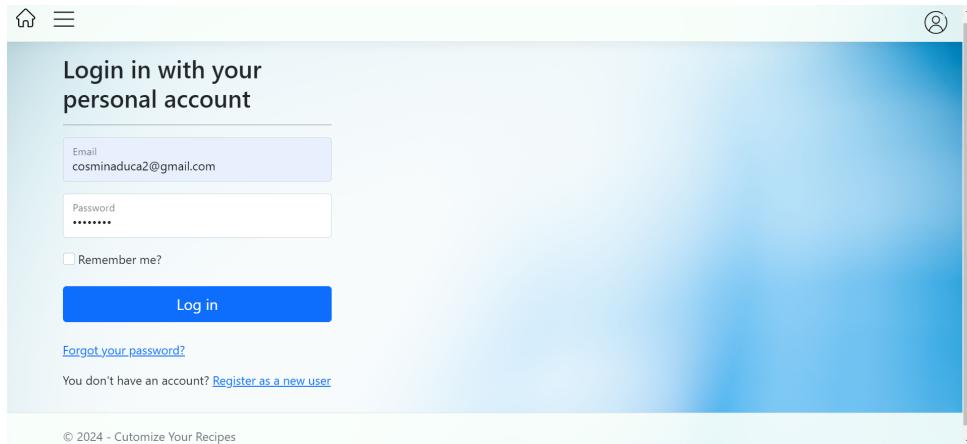


Figura 4.3: Procesul de autentificare

#### 4.2.2 Pagina Principală

Utilizatorul are în pagina principală un buton care îl trimită direct spre pagina de generare a rețetelor. În afară de asta, pagina principală conține doar o imagine de fundal în culorile alb și verde pentru nu stimula ochiul prea mult și ca butonul să fie vizibil și atrăgător.

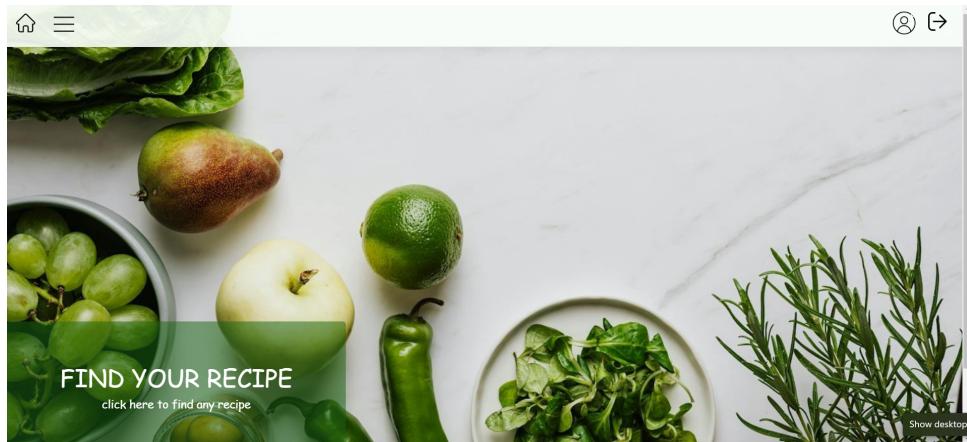


Figura 4.4: Pagina principală

#### 4.2.3 Pagina pentru Generare

În această pagină, user-ul vede secțiunea de introducere a ingredientelor, butonul pentru adăugarea unui nou ingrediente și butonul către pagina cu rețeta generată. Dacă

utilizatorul încearcă să genereze o rețetă, fără să introducă niciun ingredient, se afișează un mesaj sugestiv.

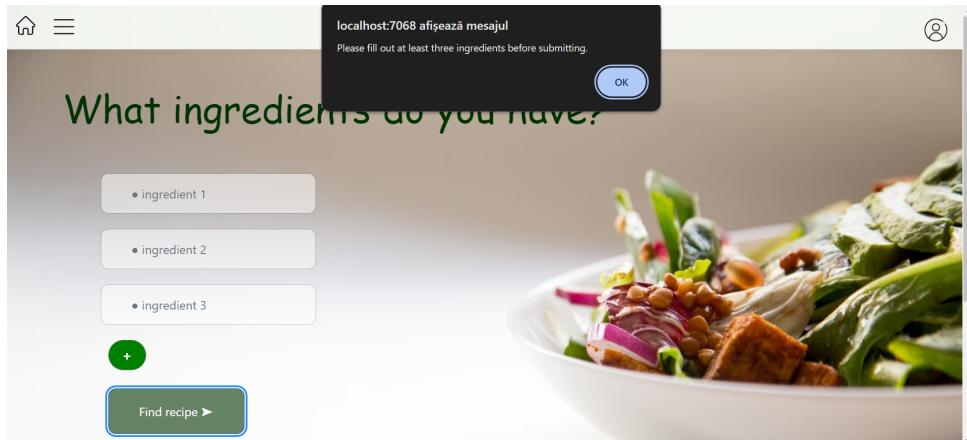


Figura 4.5: Procesul de generare nevalidat

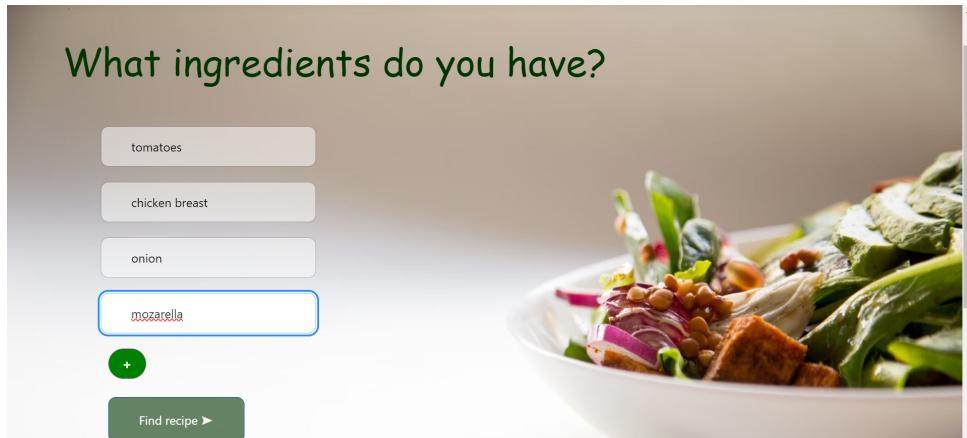


Figura 4.6: Procesul de generare validat

#### 4.2.4 Pagina Rețetei

În această pagină, utilizatorul poate citi rețeta generată cu ingredientele sale, poate vedea sugestia de prezentare în imaginea generată și poate salva rețeta, dacă își dorește.

De asemenea, tot pe această pagină utilizatorul are butonul "Save Recipe", pe care dacă îl apasă, este redirectionat pe pagina cu rețetele salvate.

#### 4.2.5 Pagina de Știri

În această pagină, user-ul poate vedea cele mai noi materiale despre prevenirea risipei alimentare și le poate accesa dând click pe thumbnail, fiind trimis pe platforma YouTube.

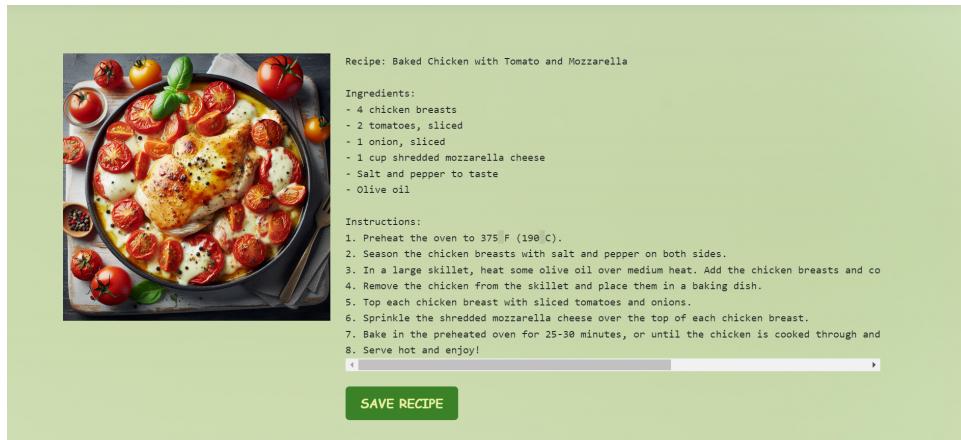


Figura 4.7: Rețeta generată

Saved Recipes

Pui cu sos de rosii si ceapa

Ingrediente:

- 500g piept de pui
- 4 rosii mari
- 2 cepe
- 2 cetei de usturoi
- 2 linguri de ulei de masline
- sare si piper
- patrunjel proaspat

Mod de preparare:

1. Curata ceapa si usturoiul si taie-le marunt.
2. Taie pieptul de pui in bucati potrivite.
3. Incinge uleiul intr-o tigarie adanca si calestie ceapa si usturoiul pana devin aurii.
4. Adauga bucatile de pui si rumeneste-le pe toate partile.
5. Taie rosurile cubulete si adauga-le peste pui.
6. Condimenteaza cu sare si piper dupa gust si lassa sa fierba la foc cu un timp de aproximativ 10 de minute.

Figura 4.8: Pagina rețetelor salvate

#### 4.2.6 Pagina Listelor de Cumpărături

În această pagină, user-ul își poate gestiona listele de cumpărături create. Poate adăuga altele noi sau le poate edita pe cele vechi, bifând, ștergând sau adăugând elemente noi.

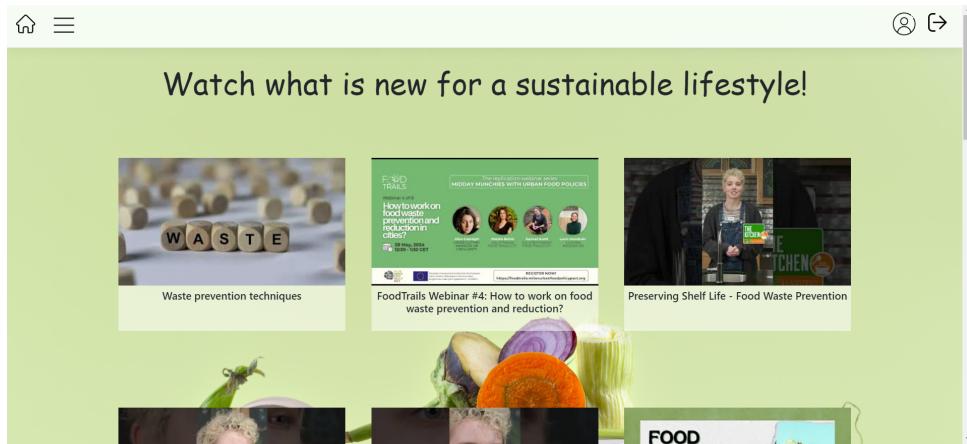


Figura 4.9: Pagina de știri

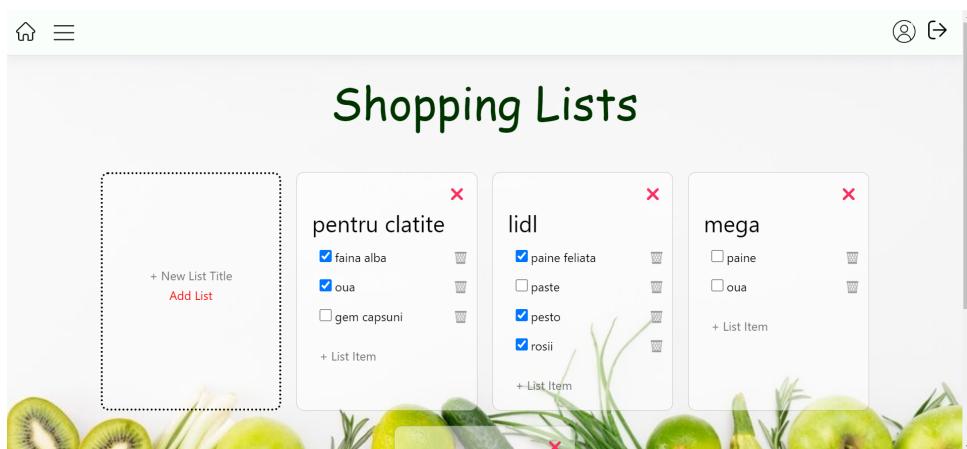


Figura 4.10: Gestionaarea listelor de cumpărături

#### 4.2.7 Pagina pentru Reciclare

În această pagină, user-ul poate căuta centrele de reciclare de care are nevoie. Aplicația preia locația acestuia și se pot urmări pe hartă cele mai apropiate centre de reciclare de la locația sa.

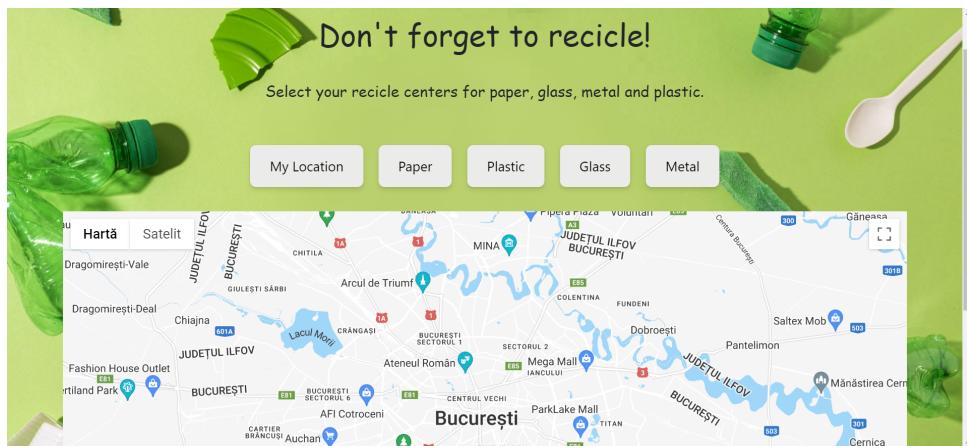


Figura 4.11: Pagina centrelor de reciclare

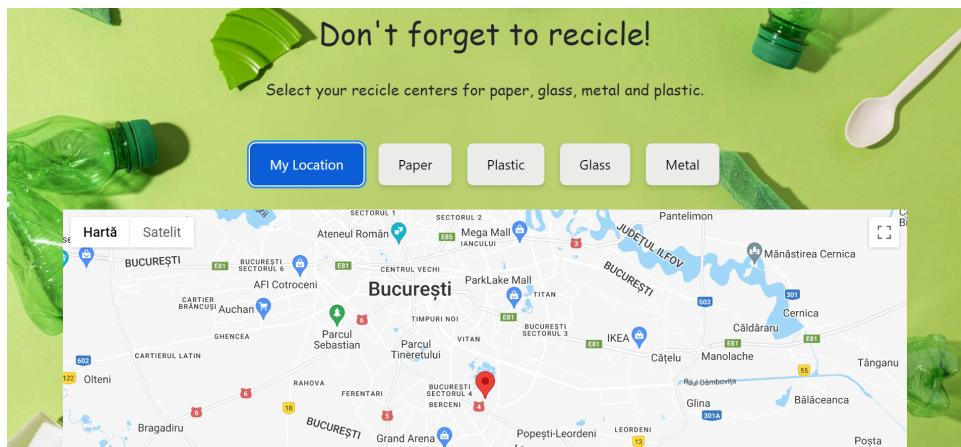


Figura 4.12: Reluarea locației utilizatorului

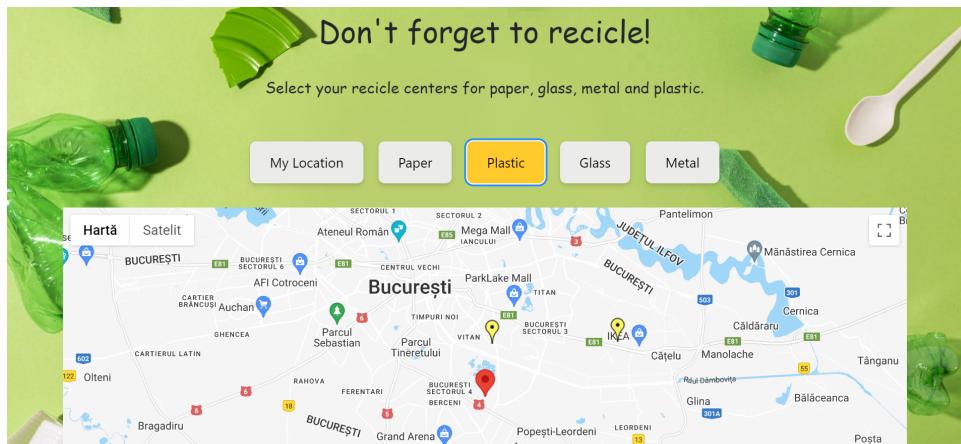


Figura 4.13: Afisarea centrelor care recicleaza hârtie

# Capitolul 5

## Concluzii

În această documentație, am analizat și am detaliat funcționarea aplicației "CookIt", axându-mă pe tehnologiile utilizate, implementarea funcționalităților și impactul acestora. Aplicația utilizează tehnici avansate precum ChatGPT pentru generarea rețetelor și DALL-E 3 pentru generarea imaginilor.

Aplicația oferă utilizatorilor o unealtă pentru gestionarea alimentelor, prevenirea risipei și promovarea reciclării, integrând funcționalități educative dar și practice. Ca un ultim pas în documentarea acestei aplicații și care vine ca o consecință a acestei lucrări, voi detalia câteva aspecte care pot fi îmbunătățite în viitor la aplicație.

**Îmbunătățirea Algoritmului de Generare a Rețetelor** - În vederea optimizării experienței utilizatorilor, se poate implementa un algoritm avansat de machine learning capabil să analizeze în detaliu preferințele culinare și comportamentul fiecărui utilizator. Acest algoritm va avea capacitatea de a învăța din interacțiunile anterioare ale utilizatorilor și va sugera rețete personalizate, adaptate gusturilor și nevoilor individuale, oferind astfel recomandări mult mai precise, relevante și interesante.

**Expandarea hărții de reciclare** - Pentru a extinde utilitatea aplicației, se va mări considerabil baza de date cu centre de reciclare, depășind granițele orașului București. Ar trebui să fie adăugate noi locații din diverse regiuni, astfel încât utilizatorii din alte zone geografice să poată beneficia de informații utile despre reciclare. Sau se poate renunța complet la baza de date, aplicația preluând din bazde de date naționale acest gen de informații. De asemenea, ar putea fi introduse noi criterii de reciclare, permitând o clasificare mai detaliată a deșeurilor.

**Crearea unei cămăre virtuale** - Se poate dezvolta o funcționalitate avansată de cămară virtuală, care va permite utilizatorilor să adauge și să gestioneze ingredientele pe care le au acasă. Aceasta va reține dățile de expirare (aproximative) ale tuturor alimentelor.

telor și astfel va putea notifica automat utilizatorii cand produsele urmează să expire.

**Transformarea într-o aplicație socială** - Pentru a îmbunătăți interacțiunea între utilizatori, aplicația va integra funcționalități sociale, permitând astfel partajarea rețeteelor și a experiențelor culinare direct între utilizatorii. Aceștia își vor putea posta rețetele preferate, comenta și aprecia postările altor utilizatori, creând astfel o comunitate activă și interactivă.

Prin implementarea acestor îmbunătățiri, aplicația "CookIt" poate deveni mai utilă și mai atraktivă pentru utilizatori, oferind funcționalități avansate și o experiență îmbunătățită. Aceste inițiative nu doar că vor contribui la eficientizarea gestionării alimentelor și la reducerea risipelui alimentare, dar vor încuraja și comportamente mai sustenabile și responsabile din partea utilizatorilor.

# Bibliografie

- [1] Bootstrap, *Introduction to Bootstrap*, Accesat: 10.06.2024, 2023, URL: <https://getbootstrap.com/docs/5.3/getting-started/introduction/>.
- [2] Coursera, *What Is UI Design? Definition, Tips, Best Practices*, (accesat în 7.6.2023), 2023, URL: <https://www.coursera.org/articles/ui-design>.
- [3] GeeksforGeeks, *Benefit of using MVC*, Accesat: 10.06.2024, 2024, URL: <https://www.geeksforgeeks.org/benefit-of-using-mvc/>.
- [4] Google Developers, *Google Maps JavaScript API - Overview*, Accesat: 10.06.2024, 2024, URL: <https://developers.google.com/maps/documentation/javascript/overview>.
- [5] Google Developers, *YouTube Data API - Getting Started*, Accesat: 10.06.2024, 2023, URL: <https://developers.google.com/youtube/v3/getting-started>.
- [6] Hotjar, *8 Ways to Improve UX Design With AI (With Recommended Tools)*, Accesat: 10.06.2024, 2024, URL: <https://www.hotjar.com/ux-design/>.
- [7] Mark Maloof, *Artificial Intelligence: An Introduction*, p. 37, <https://georgetown.edu>, Arhivat din original (PDF) la 25 august 2018, 2018, URL: <https://georgetown.edu>.
- [8] Microsoft, *Entity Framework Core*, Accesat: 10.06.2024, 2023, URL: <https://learn.microsoft.com/en-us/ef/core/>.
- [9] Microsoft, *Introduction to Identity on ASP.NET Core*, Accesat: 10.06.2024, 2024, URL: <https://learn.microsoft.com/en-us/aspnet/core/security/authentication/identity?view=aspnetcore-8.0&tabs=visual-studio>.
- [10] Microsoft, *What is ASP.NET Core?*, Accesat: 10.06.2024, 2024, URL: <https://dotnet.microsoft.com/en-us/learn/aspnet/what-is-aspnet-core>.
- [11] Microsoft Support, *Adăugarea unei conexiuni de date la o bază de date Microsoft Access*, Accesat: 10.06.2024, 2023, URL: <https://support.microsoft.com/ro-ro/office/ad%C4%83ugarea-unei-conexiuni-de-date-la-o-baz%C4%83-de-date-microsoft-access-3897bf5d-59ef-44a2-a0ca-8c41f60be390>.

- [12] Microsoft Support, *Add a Database as a Data Source*, Accesat: 10.06.2024, 2024, URL: <https://support.microsoft.com/ro-ro/office/add-a-database-as-a-data-source-3a563b0c-75e4-4856-847f-9a8182708c66>.
- [13] OpenAI, *DALL·E 2: Creating Images from Text*, Accesat: 10.06.2024, 2022, URL: <https://openai.com/index/dall-e-2/>.
- [14] OpenAI, *OpenAI API*, Accesat: 10.06.2024, 2023, URL: <https://platform.openai.com>.
- [15] OpenAI, *OpenAI API - Images Guide*, Accesat: 10.06.2024, 2024, URL: <https://platform.openai.com/docs/guides/images?context=python>.
- [16] OpenAI, *OpenAI API - Text Generation: Reproducible Outputs*, Accesat: 10.06.2024, 2024, URL: <https://platform.openai.com/docs/guides/text-generation-reproducible-outputs>.
- [17] ReFED, *Food Waste Solutions*, Accesat: 10.06.2024, 2024, URL: <https://refed.org/food-waste/the-solutions/#solutions>.
- [18] Joydip Thirumalai, *How to work with EF Core migrations in ASP.NET Core*, Accesat: 10.06.2024, 2023, URL: <https://www.infoworld.com/article/3691114/how-to-work-with-ef-core-migrations-in-aspnet-core.html>.
- [19] United Nations Environment Programme, *Food Waste Index Report 2024. Think Eat Save: Tracking Progress to Halve Global Food Waste*, Accesat: 10.06.2024, 2024, URL: <https://wedocs.unep.org/20.500.11822/45230>.
- [20] Ashish Vaswani, Noam M. Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser și Illia Polosukhin, „Attention is All you Need”, în *Neural Information Processing Systems*, 2017, URL: <https://api.semanticscholar.org/CorpusID:13756489>.
- [21] Staff Writer, *Supermarket Food Waste: Impact Solutions*, Accesat: 10.06.2024, 2023, URL: <https://shrinkthatfootprint.com/supermarket-food-waste-impacts-solutions/>.